

BBBBBBBBBBBBBBB      AAAAAAAA      SSSSSSSSSSSS      RRRRRRRRRRRRR      TTTTTTTTTTTTTTTT      LLL  
BBBBBBBBBBBBBBB      AAAAAAAA      SSSSSSSSSSSS      RRRRRRRRRRRRR      TTTTTTTTTTTTTTTT      LLL  
BBBBBBBBBBBBBBB      AAAAAAAA      SSSSSSSSSSSS      RRRRRRRRRRRRR      TTTTTTTTTTTTTTTT      LLL  
  
BBB      BBB      AAA      AAA      SSS  
  
BBBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSS  
BBBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSS  
BBBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSS  
  
BBB      BBB      AAAAAAAAAAAAAA      SSS  
BBB      BBB      AAAAAAAAAAAAAA      SSS  
BBB      BBB      AAAAAAAAAAAAAA      SSS  
BBB      BBB      AAA      AAA      SSS  
BBB      BBB      AAA      AAA      SSS  
BBB      BBB      AAA      AAA      SSS  
  
BBBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSSSS  
BBBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSSSS  
BBBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSSSS

\*\*FILE\*\*ID\*\*BASBUFSIZ

G 15

BBBBBBBB	AAAAAA	SSSSSSS	BBBBBBBB	UU	UU	FFFFFFF	SSSSSSS		ZZZZZZZZ
BBBBBBBB	AAAAAA	SSSSSSS	BBBBBB	UU	UU	FF	SSSSSS		ZZZZZZZZ
BB	BB	AA	AA	BB	UU	FF	SS		ZZ
BB	BB	AA	AA	BB	UU	FF	SS		ZZ
BB	BB	AA	AA	BB	UU	FF	SS		ZZ
BB	BB	AA	AA	BB	UU	FF	SS		ZZ
BBBBBBBB	AA	AA	SSSSS	BBBBBB	UU	FFFFF	SSSSS		ZZ
BBBBBBBB	AA	AA	SSSSS	BBBBBB	UU	FFFFF	SSSSS		ZZ
BB	BB	AAAAAAA	SS	BB	UU	FF	SS		ZZ
BB	BB	AAAAAAA	SS	BB	UU	FF	SS		ZZ
BB	BB	AA	AA	BB	UU	FF	SS		ZZ
BB	BB	AA	AA	BB	UU	FF	SS		ZZ
BBBBBBBB	AA	AA	SSSSSS	BBBBBB	UUUUUUUUUU	FF	SSSSSS		ZZZZZZZZ
BBBBBBBB	AA	AA	SSSSSS	BBBBBB	UUUUUUUUUU	FF	SSSSSS		ZZZZZZZZ
LL		SSSSSSS	SSSSSSS						
LL		SS	SS						
LL		SS	SS						
LL		SS	SS						
LL		SS	SS						
LL		SS	SS						
LLLLLLLLL		SSSSSSS	SSSSSSS						
LLLLLLLLL		SSSSSSS	SSSSSSS						

```
1 0001 0 MODULE BASSBUFSIZ (
2 0002 0 IDENT = '1-003'
3 0003 0 )
4 0004 1 BEGIN
5 0005 1 ****
6 0006 1 *
7 0007 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
8 0008 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
9 0009 1 *
10 0010 1 * ALL RIGHTS RESERVED.
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
17 0017 1 * TRANSFERRED.
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
21 0021 1 * CORPORATION.
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
25 0025 1 *
26 0026 1 *
27 0027 1 ****
28 0028 1 *
29 0029 1 *
30 0030 1 ++
31 0031 1 FACILITY: VAX-11 BASIC Miscellaneous I/O
32 0032 1
33 0033 1 ABSTRACT:
34 0034 1
35 0035 1 This module contains the BASIC BUFSIZ function, which returns
36 0036 1 the buffer size of the file open on the specified channel.
37 0037 1
38 0038 1 ENVIRONMENT: VAX-11 User Mode
39 0039 1
40 0040 1 AUTHOR: John Sauter, CREATION DATE: 11-APR-1979
41 0041 1
42 0042 1 MODIFIED BY:
43 0043 1
44 0044 1 1-001 - Original.
45 0045 1 1-002 - Set up ISBSA_USER FP. JBS 25-JUL-1979
46 0046 1 1-003 - Use channel 0, not device TT.
47 0047 1 JBS 11-MAR-1980
48 0048 1 --
49 0049 1
50 0050 1 !<BLF/PAGE>
```

```
52      0051 1 | SWITCHES:  
53      0052 1 |  
54      0053 1 |  
55      0054 1 |  
56      0055 1 | SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);  
57      0056 1 |  
58      0057 1 |  
59      0058 1 | LINKAGES:  
60      0059 1 |  
61      0060 1 |  
62      0061 1 | REQUIRE 'RTLIN:OTSLNK';           ! Define linkages  
63      0490 1 |  
64      0491 1 |  
65      0492 1 | TABLE OF CONTENTS:  
66      0493 1 |  
67      0494 1 |  
68      0495 1 | FORWARD ROUTINE  
69      0496 1 | BASSBUFSIZ;                  ! Return buffer size  
70      0497 1 |  
71      0498 1 |  
72      0499 1 | INCLUDE FILES:  
73      0500 1 |  
74      0501 1 |  
75      0502 1 | REQUIRE 'RTLML:OTSLUB';          ! Get LUB definitions  
76      0642 1 |  
77      0643 1 | REQUIRE 'RTLML:OTSISB';          ! Get ISB definitions  
78      0811 1 |  
79      0812 1 | REQUIRE 'RTLIN:RTLPSECT';        ! Macros for defining psects  
80      0907 1 |  
81      0908 1 | LIBRARY 'RTLSTARLE';           ! System symbols  
82      0909 1 |  
83      0910 1 |  
84      0911 1 | MACROS:  
85      0912 1 |  
86      0913 1 |     NONE  
87      0914 1 |  
88      0915 1 | EQUATED SYMBOLS:  
89      0916 1 |  
90      0917 1 |     NONE  
91      0918 1 |  
92      0919 1 | PSECTS:  
93      0920 1 |  
94      0921 1 | DECLARE_PSECTS (BAS);          ! Declare psects for BASS facility  
95      0922 1 |  
96      0923 1 | OWN STORAGE:  
97      0924 1 |  
98      0925 1 |     NONE  
99      0926 1 |  
100     0927 1 | EXTERNAL REFERENCES:  
101     0928 1 |  
102     0929 1 |  
103     0930 1 | EXTERNAL ROUTINE  
104     0931 1 |     BASS$OPEN ZERO : NOVALUE,  
105     0932 1 |     BASS$CB_P0SH : JSB CB PUSH NOVALUE,  
106     0933 1 |     BASS$CB_P0P : JSB CB_POP NOVALUE,  
107     0934 1 |     BASS$STOP : NOVALUE;  
108     0935 1 |  
                                ! Open channel 0  
                                ! Load register CCB  
                                ! Done with register CCB  
                                ! Signal fatal error
```

109      0936 1 | +  
110      0937 1 | - The following are the error codes used in this module.  
111      0938 1 | -  
112      0939 1 | -  
113      0940 1 | EXTERNAL LITERAL  
114      0941 1 | BASSK\_PROLOSSOR : UNSIGNED (8);                    ! Program lost, sorry  
115      0942 1 | -

```
117      0943 1 GLOBAL ROUTINE BASSBUFSIZ (           ! Return buffer size
118          0944 1   CHAN                         ! Channel whose buffer size to return
119          0945 1   ) =
120          0946 1
121          0947 1
122          0948 1
123          0949 1
124          0950 1
125          0951 1
126          0952 1
127          0953 1
128          0954 1
129          0955 1
130          0956 1
131          0957 1
132          0958 1
133          0959 1
134          0960 1
135          0961 1
136          0962 1
137          0963 1
138          0964 1
139          0965 1
140          0966 1
141          0967 1
142          0968 1
143          0969 1
144          0970 1
145          0971 1
146          0972 1
147          0973 1
148          0974 1
149          0975 1
150          0976 2
151          0977 2
152          0978 2
153          0979 2
154          0980 2
155          0981 2
156          0982 2
157          0983 2
158          0984 2
159          0985 2
160          0986 2
161          0987 2
162          0988 2
163          0989 2
164          0990 2
165          0991 2
166          0992 2
167          0993 2
168          0994 2
169          0995 2
170          0996 2
171          0997 2
172          0998 2
173          0999 3
+
++ FUNCTIONAL DESCRIPTION:
    Returns the size of the buffer for the specified channel.
    If the channel is closed a zero is returned.

FORMAL PARAMETERS:
    CHAN.rl.v      The channel whose buffer size to return.

IMPLICIT INPUTS:
    The LUBSW_RBUF_SIZE field of the LUB of the specified channel.

IMPLICIT OUTPUTS:
    NONE

ROUTINE VALUE:
    The number of bytes in the buffer, as a longword integer.

SIDE EFFECTS:
    Signals if an error is encountered.
    BAS$SCB_PUSH will signal if the channel number is invalid.

-- BEGIN
BUILTIN
FP:
GLOBAL REGISTER
  CCB = K_CCB_REG : REF BLOCK [, BYTE];
LOCAL
  BUFFER_SIZE,
  FMP : REF BLOCK [, BYTE];
  FMP = .FP;
+ If this is channel zero, get the user's terminal, and be sure it is open.
- IF (.CHAN EQ 0)
THEN
BEGIN
  BAS$SCB PUSH (LUB$K_LUN_INPU, LUB$K_ILUN_MIN);
  CCB [ISBSA_USER_FP] = .FMP [SF$L_SAVE_FP];
IF ( NOT .CCB [LUB$V_OPENED]) THEN BAS$OPEN_ZERO (.FMP [SF$L_SAVE_FP]);
```

```

174      1000 3
175      1001 2
176      1002 2
177      1003 2
178      1004 2
179      1005 2
180      1006 2
181      1007 2
182      1008 2
183      1009 2
184      1010 2
185      1011 2
186      1012 2
187      1013 2
188      1014 2
189      1015 2
190      1016 2
191      1017 2
192      1018 2
193      1019 2
194      1020 2
195      1021 1

      END
      ELSE
        BEGIN
          BASS$CB PUSH (.CHAN, LUB$K_LUN_MIN);
          CCB [ISBSA_USER_FP] = .FMP-[SF$L_SAVE_FP];
        END;

      !+ Get the buffer size from the Logical Unit Block. This will be zero
      !- if the channel has not been opened.

      !- BUFFER_SIZE = .CCB [LUB$W_RBUF_SIZE];

      !+ We are done with register CCB.
      !- BASS$CB_POP ();

      !+ All done.
      !- RETURN (.BUFFER_SIZE);

      END;

```

: end of BASSBUFSIZ

```

.TITLE BASSBUFSIZ
.IDENT \1-003\

.EXTRN BASS$OPEN ZERO, BASS$CB_PUSH
.EXTRN BASS$CB_POP, BASS$STOP
.EXTRN BASS$PROLOGOR

.PSECT _BASS$CODE,NOWRT, SHR, PIC,2

```

			081C 00000	.ENTRY	BASSBUFSIZ, Save R2,R3,R4,R11	0943
		54 00000000G	00 9E 00002	MOVAB	BASS\$CB_PUSH, R4	
		53	5D D0 00009	MOVL	FP, FMP	0988
			04 AC D5 0000C	TSTL	CHAN	0993
			1E 12 0000F	BNEQ	1\$	
		50	08 CE 00011	MNEGL	#8, R0	0996
		52	07 CE 00014	MNEGL	#7, R2	
			64 16 00017	JSB	BASS\$CB_PUSH	
FF4C	CB	0C	A3 D0 00019	MOVL	12(FMP), -180(CCB)	0997
	1A	FC	AB E8 0001F	BLBS	-4(CCB), 2\$	0999
		0C	A3 DD 00023	PUSHL	12(FMP)	
		000000006	00 01 FB 00026	CALLS	#1, BASS\$OPEN_ZERO	
			0E 11 0002D	BRB	2\$	0993
			50 D4 0002F	1\$: CLRL	R0	1004
		52	04 AC D0 00031	MOVL	CHAN, R2	
			64 16 00035	JSB	BASS\$CB_PUSH	
FF4C	CB	0C	A3 D0 00037	MOVL	12(FMP), -180(CCB)	1005
	52	D2	AB 3C 0003D	MOVZWL	-46(CCB), BUFFER_SIZE	1012
		000000006	00 16 00041	JSB	BASS\$CB_POP	1016
			52 D0 00047	MOVL	BUFFER_SIZE, R0	1020
			04 0004A	RET		1021

; Routine Size: 75 bytes. Routine Base: \_BASS\$CODE + 0000

BASSBUFSIZ  
1-003

M 15  
16-Sep-1984 00:02:51    VAX-11 Bliss-32 v4.0-742  
14-Sep-1984 11:54:43    [BASRTL.SRC]BASSBUFSIZ.B32;1

Page 6  
(3)

: 196        1022 1  
: 197        1023 1 END  
: 198        1024 1  
: 199        1025 0 ELUDOM

: ! end of module BASSBUFSIZ

#### PSECT SUMMARY

Name	Bytes	Attributes
_BASS\$CODE	75	NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

#### Library Statistics

File	Total	Symbols	Pages Mapped	Processing Time
\$_\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	1	0	00:01.2

#### COMMAND QUALIFIERS

: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS\$:BASBUFSIZ/OBJ=OBJ\$:BASBUFSIZ MSRC\$:BASBUFSIZ/UPDATE=(ENH\$:BASBUFSIZ )

: Size:        75 code + 0 data bytes  
: Run Time:    00:08.3  
: Elapsed Time: 00:20.5  
: Lines/CPU Min: 7436  
: Lexemes/CPU-Min: 45155  
: Memory Used: 115 pages  
: Compilation Complete

0019 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY